

# Spectral Noise Estimation: A Python 3 Implementation of the Minimum Statistics Estimation

Bello N.<sup>1,\*</sup> and Ogbeide K. O.<sup>2</sup>

<sup>1,2</sup>Department of Electrical/Electronic Engineering, University of Benin, Benin City, Edo State, Nigeria

Corresponding Author: \*nosabello@uniben.edu

<https://doi.org/10.36263/nijest.2022.01.0300>

## ABSTRACT

Noise estimation has been used majorly in imaging processing and voice speech recognition applications. Therefore, researchers have found optimal solutions to non-stationary noise estimation. Particularly, there is a proposed method that estimates spectral noise in a noisy speech signal which is based on two observations; speech pauses and approximation of power spectral densities of the noisy signal to the true noise during speech pauses. Though from recent studies, the observations obtained cannot be inferred for other types of signals especially RF signals and have not been tested on signals in the frequency domain, this paper bridges that gap of research and presents the results, analysis, and conclusion on the findings concerning the noise estimation with RF signals using an extension of the proposed method in the frequency domain. It presents a detailed methodology of implementation of the minimum statistics method for noise estimation in python 3 code which was tested with RF signals and thus met the requirement of dynamic thresholding with spectrum occupancy measurement.

**Keywords:** Spectral estimation, Minimum statistics, Smoothing, Noise estimation, RF signals, Dynamic thresholding

## 1.0. Introduction

Noise variation occurs not just in frequency but in time (Series, 2017) and a preferred solution is one that is able to consider the frequency- and time- dependent noise levels in occupancy measurement. This is because a strong emission in one channel may produce an increase in the noise level in neighbouring channels due to the phase noise of the transmitter. A solution to such changing noise level suggests that if noise level is measured on each channel separately, the resulting threshold prevents phantom occupancies. Some methods have been proposed by the International Telecommunication Union (ITU) to achieve dynamic thresholding in the spectrum occupancy measurement. One of such methods uses the frequencies of an unused channel such that the threshold is calculated from the levels measured throughout one scan. Firstly, the method only works in the frequency band occupancy measurement or the occupancy measurement of multiple channels of equal bandwidth because it tracks only the frequency dependent noise levels. Secondly, it does not take into consideration that strong emissions in neighbouring channels can affect the noise level in the referenced channel. Another method called the 80% method is described in Recommendation ITUR SM.1753. According to the method, 80 per cent of all the samples representing the highest are discarded and the remaining 20 per cent of the samples representing the lower levels are linearly averaged (Qu, et al., 2014). Despite the advantage of this method over the previous method as it does not require an unused channel or idle times, the calculated noise level rises when more channels are occupied with high levels, such that the measurement sensitivity is lost. It is therefore thoughtful to implement noise estimation which is much more advance than the simple 80% method in dynamic thresholding. It shares the advantage as the 80% method but eliminates the sensitivity drawback. The objective of the method is to estimate the noise as accurately as possible irrespective of the state of the channels.

Optimal spectral noise estimation approaches exist with key components of power spectral density smoothing algorithm (Martin, 2001) which employs a time varying smoothing parameter, an algorithm to track the variance of the smoothed power spectral density in frequency bands, and a bias compensation algorithm for minimum power spectral density estimates. This algorithm showed that the time varying smoothing significantly improves the minimum statistics approach and turned out to be fairly generic with different noise types. This paper presents a nuanced version of the method of minimum statistics in noise estimation of RF signals for spectrum occupancy measurement implemented in Python 3.6.

Section 2 gives detailed review of the existing algorithm by (Martin, 2001; Martin, 2006; Mauler & Martin, 2006) and presents the corresponding results. The methodical approach taken towards the implementation of the reviewed algorithm and the application of nuances for RF spectral noise estimation in the frequency domain are presented in Section 3. Section 4 contains the research findings demonstrated through measures – the mean squared error was determined. First, we validate our implementation with that of Martins and then show how the proposed method improves the performance of the recommendations made by ITU. In Section 5 the summary and conclusions are given.

## 2.0. Review of Minimum Statistics (MS) Noise Estimation

Martin's (Martin, 2001; Martin, 2006; Mauler & Martin, 2006) method was based on minimum statistics and optimal smoothing of the noisy speech power spectral density. This method rested on two major observations. The first observation was independence of speech and noise which implied that the power spectrum of the noisy speech was equal to the sum of the power spectrum of clean speech and noise respectively.

That is,

$$Y(n,k)^2 = X(n,k)^2 + D(n,k)^2 \quad (1)$$

where  $Y(n,k)^2$ ,  $X(n,k)^2$  and  $D(n,k)^2$  were the power spectrum of noisy speech, clean speech and noise respectively and  $n$  and  $k$  denoted the time index and frequency bin index respectively. The second observation was that the power spectrum of noisy speech often becomes equal to the power spectrum of noise. This happens during speech pauses and also between words and syllables. Hence the estimate of noise power spectral density was obtained by tracking the minimum of the noisy speech in each frequency bin separately. However, the minimum was biased towards lower values, so the unbiased estimate was obtained by multiplying with a bias factor which was derived from the statistics of the local minimum.

### 2.1. Principles of minimum statistics algorithm

In theory, the power spectral density of noisy speech is equal to the sum of noise power and speech power; therefore, noise can be estimated by tracking the minimum of the noisy speech power spectral density over a fixed window length.

For searching the minimum, a first-order recursive version of the noisy speech power spectral density was used:

$$P(n,k) = \alpha P(n,k) + (1 - \alpha)|Y(n,k)|^2 \quad (2)$$

where  $\alpha$  was the smoothing constant. To improve the performance of the minimum tracking procedure the following modifications were made:

- i. Replacing the constant smoothing factor in Equation (2) with the time-frequency dependent smoothing factor.
- ii. Deriving a bias factor for the noise estimate since the minimum tracking was biased towards lower values.
- iii. Improving tracking speed of the algorithm for increasing noise levels.

## 2.2. Deriving optimal time-frequency dependent smoothing factor

The smoothing parameter used in Equation (2) had to be very small to follow the non-stationarity of the speech signal. On the other hand, it had to be close to one to keep the variance of the minimum tracking as small as possible. Hence there was need for time and frequency dependent smoothing factors in place of a fixed smoothing factor. This was derived for speech absent region. The requirement was that the smoothed power spectrum,  $P(n, k)$  had to be equal to the noise variance  $\sigma_D^2(n, k)$  during speech pauses. Hence the smoothing parameter was derived by minimizing the mean squared error between  $P(n, k)$  and  $\sigma_D^2(n, k)$  as follows:

$$E \left\{ \left( P(n, k) - \sigma_D^2(n, k) \right)^2 \mid P(n-1, k) \right\} \quad (3)$$

Where

$$P(n, k) = \alpha(n, k)P(n, k) + (1 - \alpha(n, k))|Y(n, k)|^2 \quad (4)$$

Note that in Equation (4) the time-frequency dependent smoothing factor  $\alpha(n, k)$  was used instead of fixed  $\alpha$  as defined in Equation (2). Substituting Equation (4) in (3) and setting the first derivative to zero gave the optimum value for  $\alpha(n, k)$  as:

$$\alpha_{opt}(n, k) = \frac{1}{1 + \left( \frac{P(n-1, k)}{\sigma_D^2(n, k)} - 1 \right)^2} \quad (5)$$

But in real time implementation, the value of estimated noise variance,  $\hat{\sigma}_D^2(n, k)$  lags behind true noise variance,  $\sigma_D^2(n, k)$ . Hence some correction factor  $\alpha_c(n)$  was calculated using the ratio of averaged smoothed periodogram to estimated noise variance. The final smoothing parameter after the correction factor was given as:

$$\alpha_{opt}(n, k) = \frac{\alpha_{max}\alpha_c(n)}{1 + \left( \frac{P(n-1, k)}{\hat{\sigma}_D^2(n, k)} - 1 \right)^2} \quad (6)$$

Where  $\alpha_{max} = 0.96$ .

## 2.3. Bias factor

As the minimum was biased towards lower values, the bias factor for compensating the minimum of noisy speech power spectrum was derived using the statistics of minimum of the correlated PSD estimates of noisy speech. It was stated that since probability density function (pdf) of  $P(n, k)$  was scaled by  $\sigma_D^2(n, k)$ , the minimum statistics of the short-term estimates  $P_{min}(n, k)$  was also scaled by  $\sigma_D^2(n, k)$ . Thus, the bias term was derived by finding the mean of minimum PSD for some  $\sigma_D^2(n, k) = 1$  which after simplification gave:

$$B_{min}(n, k) \approx 1 + (L - 1) \frac{2}{\tilde{Q}_{eq}(n, k)} \quad (7)$$

where  $L$  was the window length over which the minimum was found and  $\tilde{Q}_{eq}(n, k)$  called “equivalent degrees of freedom”, was a function of smoothed periodogram, and the previous noise variance. The unbiased noise estimate was then obtained as:

$$\sigma_D^2(n, k) = B_{min}(n, k)P_{min}(n, k) \quad (8)$$

It is advantageous to increase the inverse bias,  $B_{min}(n, k)$ , by a factor,  $B_c(n)$  proportional to the normalized standard deviation of the short-term estimate,  $P(n, k)$ .

$$B_c(n) = 1 + 2.12\sqrt{\bar{Q}^{-1}(n)} \quad (9)$$

with the average normalized  $\bar{Q}^{-1}(n) = \sum_k \frac{1}{Q_{eq}(n,k)}$  variance. This bias correction has an impact only when the short-term PSD estimate and thus the estimated variance has a large variance. Without the bias correction the variations in the bias factor would push the minimum to values which are too low.

#### 2.4. Efficient minimum search algorithm

The minimum for the noisy speech was found over  $L$  consecutive frames. In the worst case of increasing noise levels, the minimum search lagged behind by  $2L$  frames. To reduce this lag the whole window of  $L$  frames was divided into  $U$  sub windows of  $V$  samples each ( $L = UV$ ). In this case, the maximum delay was reduced to  $L+V$  frames compared to  $2L$  frames in the previous case.

#### 2.5. Experimental results

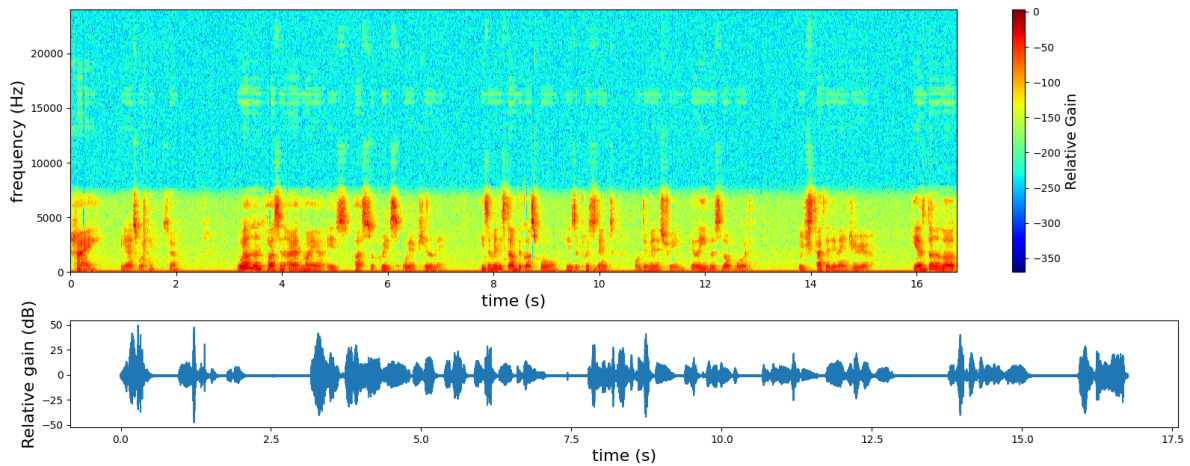
The performance of the noise estimation algorithm was assessed using both objective and subjective measures. For objective measure, the percentage relative estimation error and the error variance were calculated between the true noise spectrum and estimated noise spectrum for white Gaussian noise, vehicular noise and street noise with a SNR of 15 dB using continuous English sentences and sentences with speech pauses. For subjective analysis, the noise estimation algorithm was combined with multiplicatively modified minimum mean square error log spectral amplitude (MM-MMSE-LSA) estimator (Ephraim & Malah, 1985; Malah, et al., 1999) and the 2400bps MELP speech coder (McCree, et al., 1996). The diagnostic acceptability measure (DAM) and diagnostic rhyme test (DRT) scores were presented. All the results showed superior performance for minimum statistics approach compared to the VAD (Martin, 2001). The tracking of minimum in each frequency bin separately helped in retaining the weak voiced consonants which may be classified as noise only frames by most of the VAD's since their energy was concentrated in very small number of frequency bins.

### 3.0. Material and Methodology

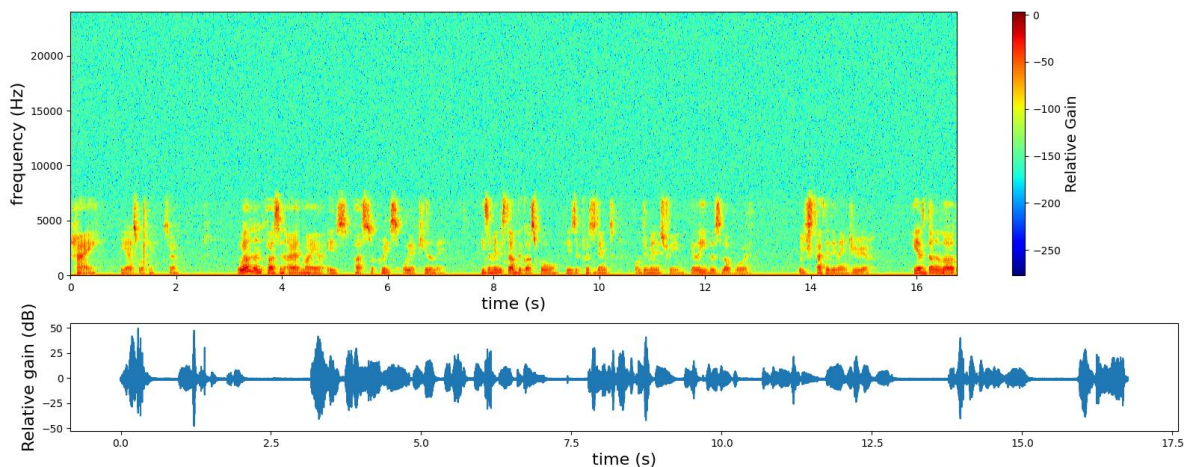
#### 3.1. Testbed for development

Briefly, we will discuss the methodology in a high-level overview of the sequence of steps or procedure taken towards the findings and results achieved in this research. To start with, we used a speech signal for the development and debugging process and then we tested the program with RF signals which were generated with GNU Radio application. GNU Radio (Seeber, 2014) is a free and open-source software development toolkit that provides signal processing blocks to implement software-defined radios and signal- processing systems. It is a highly modular and flowgraph-oriented framework that comes with a comprehensive library of processing blocks easily combined to make complex signal processing applications. It enables users to design, simulate and deploy highly capable real-world radio systems and is widely used in hobbyist, academic and commercial environments to support both wireless communications research and real-world radio systems.

The results of the speech signal were correlated with those presented by Martin and the validation of the proposed design was done using a minimum squared error analysis of the true noise and the estimated noise. The speech signal used for the development was recorded in a calm environment with a phone voice recorder application. It was 16.76 seconds long and it represented the clean speech signal, whereas the noisy speech signal was generated as an additive white Gaussian noise at an SNR of 40dB to the clean speech signal according to the theory behind the proposed method. The spectrogram plots of the clean speech and the noisy speech are shown in Figures 1 and 2 respectively. They were produced at a 25 milliseconds short-time Fourier analysis using an overlap of 50% (Mateo & Talavera, 2018). The development and debugging were done in Python 3 version.



**Figure 1:** Time spectrum and spectrogram of clean speech signal using a window time of 25 millisecond for a short time Fourier analysis



**Figure 2:** Time spectrum and spectrogram of noisy speech signal at an SNR of 40dB using a window time of 25 millisecond for a short time Fourier analysis

### 3.2. Derive the smoothing factor and the smoothed power spectrum density samples

The first step in the development is computing the smoothing factor for determining the smoothed PSD samples. Also in this stage, a decision on the best programming style was considered, of which we chose a functional and object-oriented style of development. The algorithm starts with the FFT bins of the sample window of a certain length and then calculates the smoothing parameter from Equations (5) – (6). The values of the smoothing factor are used to compute the smoothed PSD using the recursive relation in Equation (4). To verify the results, we produced the plots of the smoothed PSD vs the PSD of the noisy signal, and the smoothing factor. It is expected that;

- i) The smoothing factor should approach 0.96 at speech pauses and then approach 0 at frames containing speech signals.
- ii) Also, the smoothed noisy speech PSD should be smoother during speech pauses and less smooth when there is speech in the frames.

### 3.3. Bias factor for unbiased noise estimate based on minimum statistics

Next in the development process is the derivation of the unbiased noise estimate. To determine an unbiased noise estimate from the minimum statistics of the periodogram,  $P_{min}(n, k)$ , a bias factor is needed. The sequence of steps to compute the bias,  $B_{min}(n, k)$ , uses Equation (7). To start with, the smoothing parameter for determining the first and second moment of the smoothed PSD is determined as they provide an approximate of the variance. The bias factor is dependent on the length of the correlated smoothed PSD,  $P_{min}(n, k)$ , and the equivalent degree of freedom which in turn is a factor of the variance and the previous estimated noise. The correction factor is also added to affect large variance in the estimated noise.

### 3.4. Nuanced noise estimation in the frequency domain

It has been demonstrated how efficient the noise estimation based on minimum statistics is in the time domain of the signal. However, it truly scientific to be curious of the effect of the noise estimation based on minimum statistics in the frequency domain since spectrum occupancy measurement considers the frequency domain in its evaluation. So, we explored the implemented algorithm in the frequency domain first observing the performance with the speech signal, then with the RF signals. The proposed changes addressed the computation of the smoothed PSD and the unbiased noise estimate. To heighten the estimation of the unbiased noise in the frequency domain, the smoothed PSD should not only smoothen the time domain but also the frequency domain. Hence, equation (4) which smoothen the PSD of the noisy speech signal along the time domain (the time index  $n$ ) was changed to smoothen in both the time and frequency domain. Prior to the changes, the equation computes the newest time vector of the smoothed PSD using the previous time vectors – smoothed PSD and the smoothing factor – this means it is recursive in time. To ensure the recursion is in both time and frequency, we made the recursion sweep through the previous time vector from each frequency component to the next while smoothing each PSD sample. Also, to acknowledge the unbiased noise estimate in the frequency domain, the same steps taken in noise estimation in the time domain were repeated in the frequency domain. However, the use of sub-window in the algorithm was not needed as the frequency index is usually a small value dependent on the user's preference when analysing signals. Therefore, the length of the window used during the program would be used to track the minimum of the frequency samples. The pseudo code that shows the minimum statistics tracking in the frequency domain is given in Algorithm 1.

---

#### Algorithm 1: Pseudo code for minimum statistics noise estimation in the frequency domain

---

For  $k$  in *frequency\_bins*:

if  $actmin(n, k) \times noise\_slope\_max > min(P(n, k) \times B_{min}(n, k) \times B_c(n))$ :  
      $actmin(n, k) = min(P(n, k) \times B_{min}(n, k) \times B_c(n))$   
      $noise\_est(n, k) = actmin(n, k)$

---

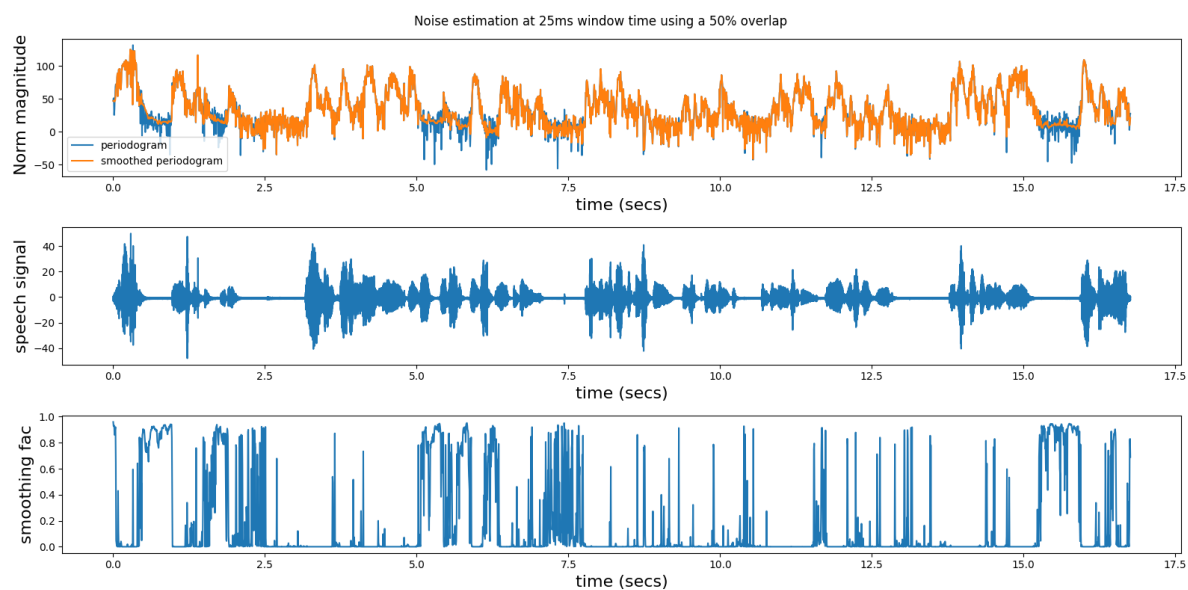
The algorithm shows the moving tracker,  $actmin(n, k)$ , which holds the minimum of the window of length  $L$ . It is only updated whenever the minimum of the newest window is less than the product minimum tracker and the  $noise\_slope\_max$ . Just like the algorithm proposed by Martin, the  $noise\_slope\_max$  is the allowance given to the existing minimum before it is updated.

The study also proceeded towards testing the proposed algorithm with RF signals. The RF signal dataset used for algorithm testing and comparison was generated with GNU Radio application. The dataset consists of 6 channels that was a mix of different signals – orthogonal frequency division multiplexing (OFDM), binary phase shift keying (BPSK), and frequency Modulation (FM) signals. The spectrum band had two OFDM signals, two FM signals, a BPSK signal and a free channel which is located between the BPSK and the FM signals. The free channel in the data set was used to ensure that we could compare the performance of our proposed method and the two existing noise estimation methods proposed by ITU.

Lastly, we tested the proposed algorithm with measured RF signals in the FM band (87.5 – 108 MHz) to show the results of spectrum occupancy measurement (SOM).

#### 4.0. Results and Discussion

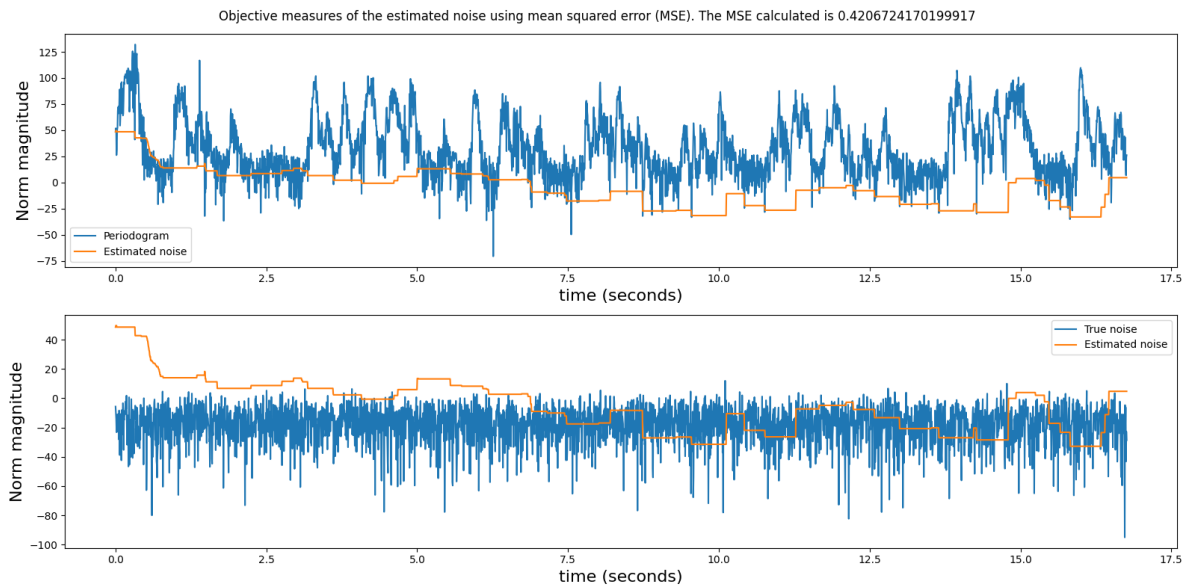
The spectral noise estimation program was a full implementation in Python 3. Prior to this paper, it was difficult to obtain an existing open-source python code program for Martin's algorithm, therefore, at the end of the research it was successfully implemented. The steps taken to successfully achieve the valid implementation of the proposed algorithm by Martin have been discussed in the section 3 and in this section, we interpret the results and compare results with that provided by Martin. First, the methodology achieved the implementation of the smoothing factor. Figure 3 shows the result of the smoothing parameter. To verify the accuracy of the results, we plotted the noisy speech signal alongside the results of the smoothed PSD vs PSD subplot (top subplot) and the smoothing factor (bottom subplot). As expected, at speech pauses, the smoothing factor attains values close to one and the smoothed PSD is smoother. Moreover, there are dips (where the smoothing factor approaches zero) when the speech is present and the smoothed PSD samples are less smooth.



**Figure 3:** Three subplots showing the computed PSD samples and the PSD samples, the time spectrum of the noisy speech signal and the smoothing factor

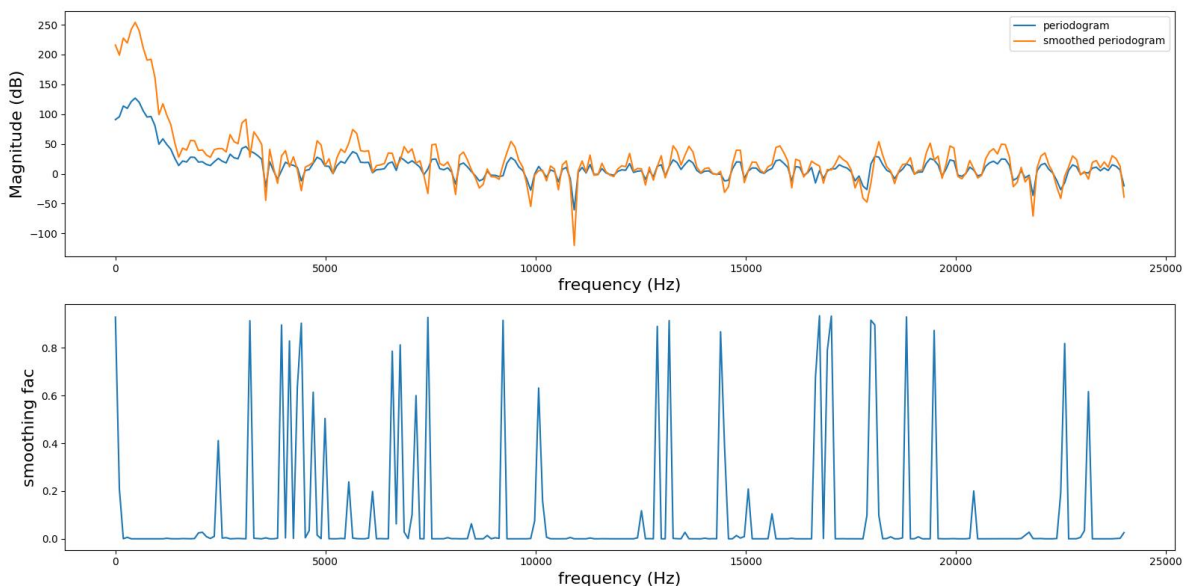
With the first step solved, the next and final step is the computation of the unbiased estimate noise samples whose result is presented in Figure 4. It shows the subplots of the noise estimate on the noisy speech signal and the true noise versus the estimated noise. The figure shows two subplots – the noise estimate versus the noisy speech signal and the true noise versus the estimated noise. The top plot shows that the estimated noise attains higher levels at the start compared to the end and based on the results of the bottom plot the estimation is closer to the true noise towards the end of the plot.





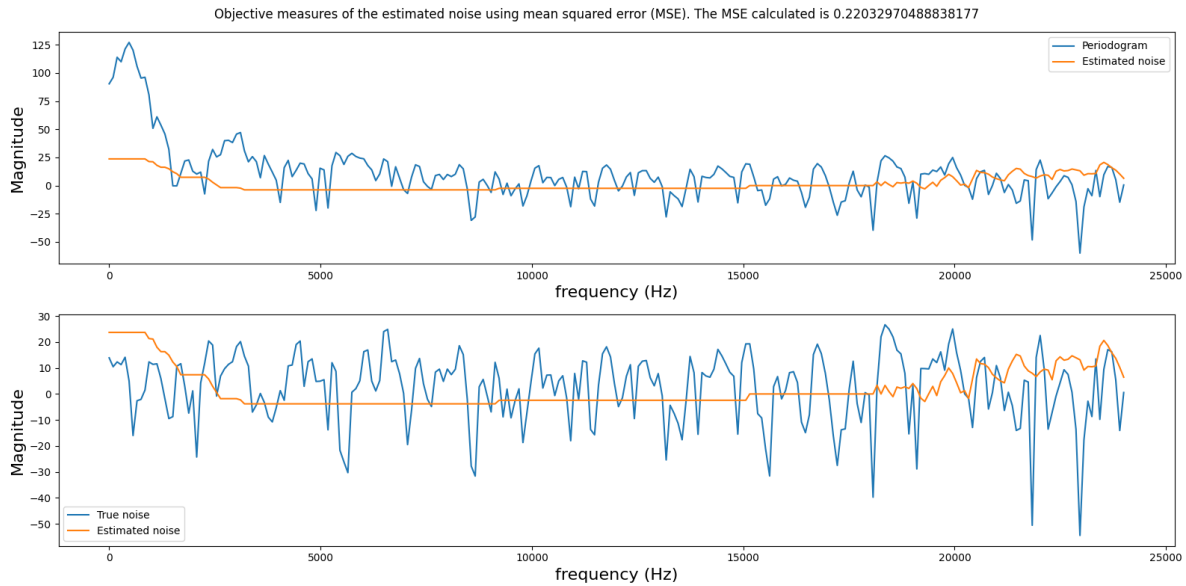
**Figure 4:** Estimation validations of algorithm with subplots showing the noisy speech periodogram versus the estimated noise and the true noise versus the estimated noise.

As stated earlier, Martin performed experimental analysis using the percentage relative estimation error and the error variance between the noise estimate and the true noise for the different noise types he used. The mean squared error (MSE) is the mean of the error variance which is an acceptable metric for estimation analysis. The computed normalized MSE for the noisy speech signal using Martin's algorithm was 0.4207. Same method of analysis was used for the shaded version we proposed and the results are presented in Figures 5 and 6. They show the plots of the smoothing factor and the noise estimation. With the proposed method, the MSE performance analysis gave a value of 0.2203 for the noisy speech signal.



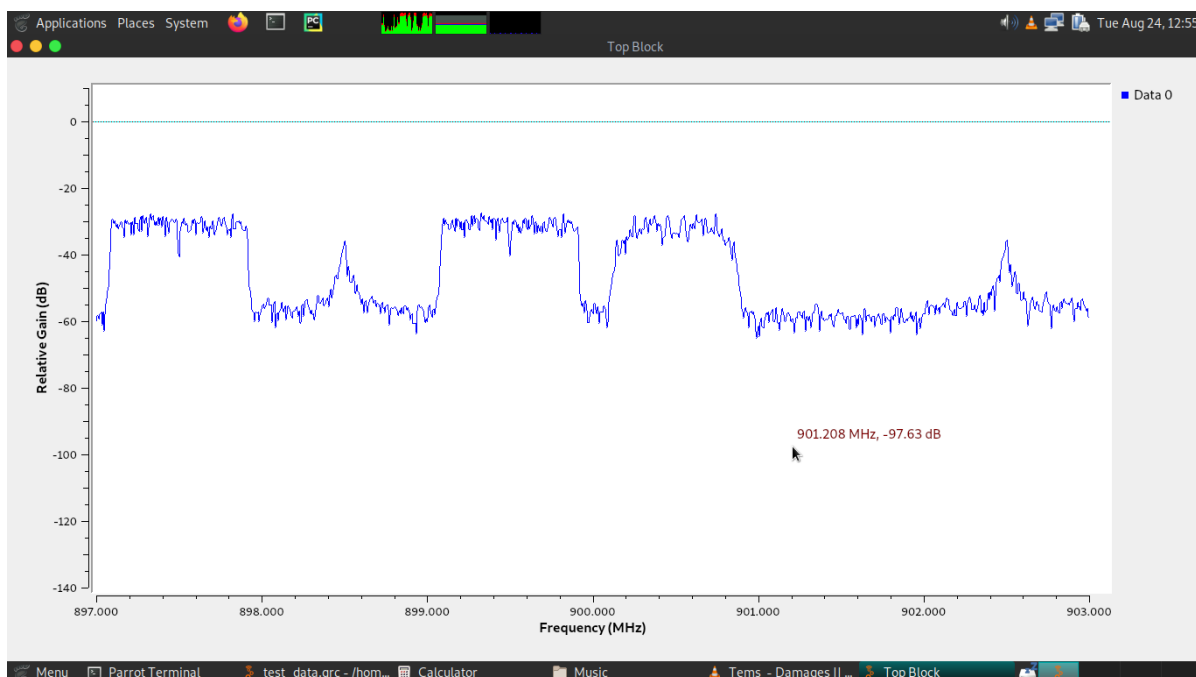
**Figure 5:** Two subplots showing the computed PSD samples versus the PSD samples and the smoothing factor





**Figure 6:** Estimation validations of algorithm with subplots showing the noisy speech periodogram versus the estimated noise and the true noise versus the estimated noise

Figures 7 and 8 shows the spectrum plot of the signals across 6 MHz bandwidth and the corresponding comma-separated values (CSV) data. The results produced with the RF signal noise estimation were expected as with that of the noisy speech signal. The noise estimation in the RF signal tends towards the noise level in the spectrum plot and is unbiased towards the minimum. In comparison to other existing methods – the free channel and the 80% methods, the noise estimation is dynamic in time and frequency and shows close approximation to the 80% method than the free channel method which completely depends on the presence of a free channel during measurement. The result is presented in Figure 9 that shows the test spectrum band having 6 channels of equal bandwidth that are transmitting at equal power. The resemblance of the estimation gotten by the 80% method and the proposed method shows that the noise estimation accuracy has been maintained.



**Figure 7:** Spectrum plot of the band of RF signals used for algorithm testing and comparison with other existing methods

Applications Places System test\_data\_raw\_csv.csv - LibreOffice Calc Wed Aug 25, 15:22

File Edit View Insert Format Styles Sheet Data Tools Window Help

LibreOffice Calc

Sheet1

	A	B	C	D	E	F	G	H	I	J	K
1	time(frequency)	-3000000	-2991202.34604106	-2982404.69208211	-2973607.03812317	-2964809.38416422	-2956011.73020528	-2947214.07624633	-2938416.42228739	-2929618.76832845	-2920821.11
2	0	0.09211655	0.07492552	0.01499598	0.11401256	0.16064192	0.07382502	0.010116715	0.05655986	0.13574706	0.187
3	1	0.10968757	0.05873894	0.18398981	0.19842993	0.17609693	0.1866356	0.18147261	0.1713882	0.23821929	0.41
4	2	0.010706606	0.12319394	0.15210019	0.14888266	0.182773	0.16394036	0.15995526	0.18933375	0.13236393	0.0320
5	3	0.103643395	0.09776035	0.165462	0.217077	0.28060326	0.32475412	0.30098918	0.15558057	0.10645863	0.264
6	4	0.27817598	0.3242353	0.32187483	0.24636304	0.22216144	0.22299531	0.14606218	0.08760708	0.26572797	0.262
7	5	0.15382372	0.13963549	0.14875542	0.14831929	0.13824366	0.122760035	0.027943237	0.14740385	0.18941027	0.177
8	6	0.1546501	0.07534776	0.103800766	0.17016506	0.12613799	0.084542066	0.19845116	0.25322407	0.23311536	0.23
9	7	0.10485263	0.10277384	0.09838326	0.0805366	0.012473247	0.11506378	0.20999832	0.2545693	0.26447573	0.202
10	8	0.06407132	0.035591684	0.027181232	0.03784139	0.074829504	0.15606132	0.19367689	0.2826283	0.40617925	0.416
11	9	0.2208132	0.2015322	0.087255195	0.03727512	0.10262927	0.1916978	0.23773241	0.22602785	0.1410654	0.156
12	10	0.09322106	0.037523527	0.057808954	0.12655602	0.12309332	0.13102888	0.2709471	0.39178005	0.40867049	0.45
13	11	0.04763831	0.16936553	0.19737092	0.16109896	0.23632073	0.34402856	0.44520432	0.43443164	0.2945511	0.207
14	12	0.19891798	0.15200318	0.08143448	0.1814563	0.13562685	0.0880289	0.14131966	0.21029198	0.2587787	0.225
15	13	0.051956303	0.106711105	0.118265964	0.18522951	0.26926355	0.20169955	0.073426	0.15915248	0.14917015	0.067
16	14	0.11222919	0.11593782	0.071440704	0.07095555	0.17734422	0.25582	0.27770594	0.26850075	0.17321223	0.095
17	15	0.15317987	0.15670006	0.18382564	0.13421081	0.09250888	0.12378283	0.0351198	0.11696961	0.14695608	0.0622
18	16	0.13493288	0.19879688	0.22119924	0.22172359	0.15852796	0.08144858	0.091413125	0.09266736	0.19904637	0.326
19	17	0.22220472	0.17576553	0.18446723	0.20658755	0.17524943	0.08121997	0.13086627	0.12416408	0.086238824	0.195
20	18	0.16399609	0.16077046	0.16176192	0.20396803	0.26045144	0.24576782	0.20649154	0.14716223	0.06937215	0.175
21	19	0.23094438	0.121250845	0.07523017	0.14691778	0.2145535	0.28951085	0.29880068	0.22490947	0.21596055	0.27
22	20	0.18377618	0.2386186	0.23111124	0.17860423	0.1670317	0.24031988	0.31584308	0.2711399	0.30486163	0.42
23	21	0.15581356	0.15841605	0.17428465	0.16563119	0.09915099	0.019377375	0.08416779	0.07949199	0.08080385	0.111
24	22	0.14390378	0.035992607	0.101516485	0.11586773	0.22140656	0.3708181	0.48699108	0.423943	0.16480821	0.03
25	23	0.06271773	0.06941763	0.07323416	0.09022026	0.04908357	0.101011954	0.16190806	0.095317885	0.10010504	0.121
26	24	0.10025275	0.11937814	0.14755432	0.16714191	0.14659433	0.057615686	0.046611097	0.08444894	0.12753575	0.195
27	25	0.12119486	0.12241622	0.16950221	0.13572642	0.059434015	0.10125487	0.046631545	0.082959905	0.19197993	0.235
28	26	0.17291331	0.1975324	0.09117233	0.09280965	0.120285906	0.071100585	0.13753389	0.22403693	0.21498029	0.175
29	27	0.1333403	0.07124173	0.018247264	0.022289743	0.1385127	0.19861072	0.19341221	0.24217908	0.29480729	0.194
30	28	0.36468614	0.16727228	0.03713074	0.04307846	0.08301459	0.12871365	0.08209776	0.11544349	0.096846975	0.06
31	29	0.10260266	0.05016279	0.09427371	0.022064763	0.09041303	0.15000445	0.1466117	0.1036308	0.17003716	0.15

test\_data\_raw\_csv

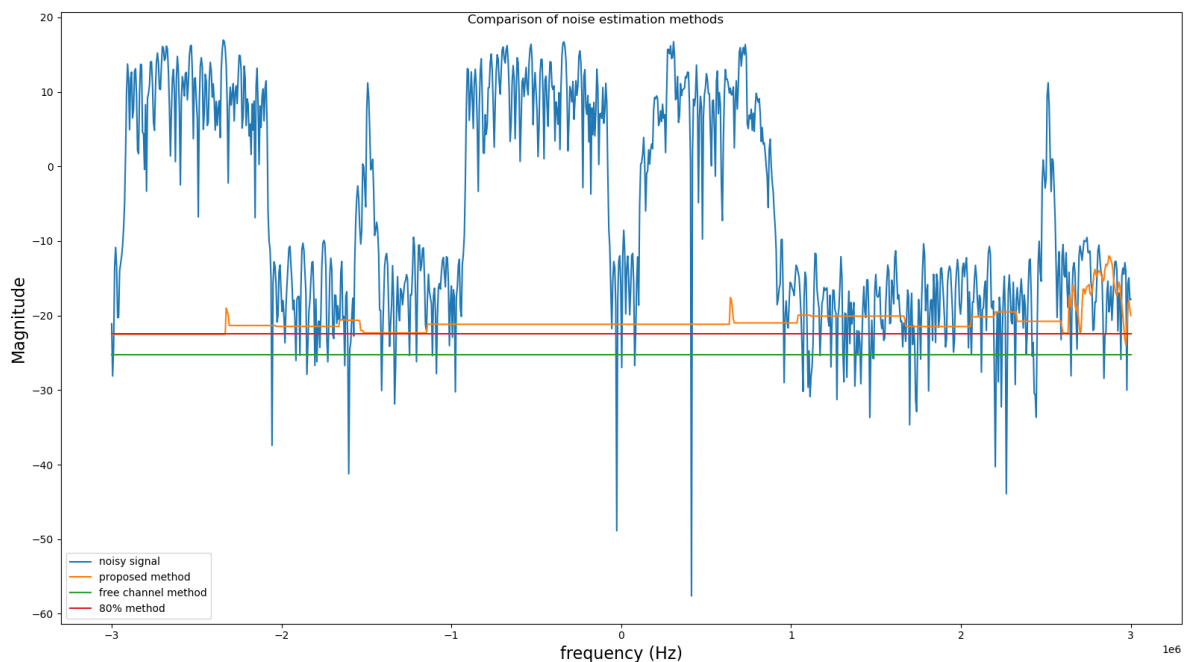
Sheet1 of 1

test\_data\_raw\_csv.csv ... Marco

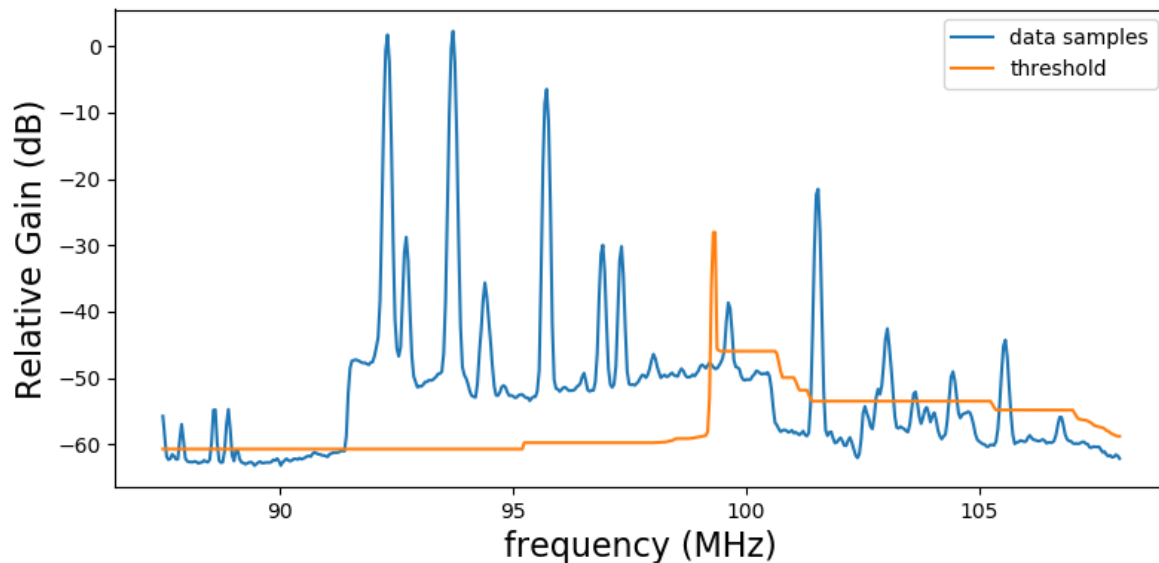
Default English (Nigeria) Average: 2.046734; Sum: 2.046734 100%

**Figure 8:** CSV data for the RF signals transmitting in 6 MHz bandwidth

In addition, we did a 15 mins test of the proposed method on two datasets measured in the FM band using GNU Radio, USRP B210s and antennas, a setup discussed in Bello and Ogbeide (2021). The FM signal test showed an occupancy of approximately 34 percent which is shown in Figure 10.



**Figure 9:** Comparison of existing methods with proposed method



**Figure 10:** Spectrum occupancy measurement in the FM band (87.5 – 108 MHz) using 3 dB threshold above the noise estimation

The results presented have shown the successful implementation of the noise estimation algorithm that uses minimum statistics and the improved noise estimation from the proposed method since the MSE is reduced from 0.4207 to 0.2203 which is approximately 50 percent. Also, the proposed noise estimation method closely correlates with the estimation using the 80 percent method however; it shows dynamism in time and frequency. Particularly in spectrum occupancy measurement that spans a large bandwidth the noise level shows appreciable changes with frequency (Turner, 2012), thus a noise estimation that tracks both changes in time and frequency has better accuracy with changing threshold values.

## 5.0. Conclusions

In the end, the successful implementation in python 3 code was achieved for noise estimation using the minimum statistics method and observations were drawn from the results which show that the method is not just peculiar to speech signals but can be inferred for RF signals as well since the same rule of independence exist between clean RF signals and noise. Also, the study presented the strengths and weaknesses of existing methods proposed by ITU and how the proposed method has contributed to improved noise estimation with dynamism in time and frequency. Such a dynamic feature improves the accuracy of spectrum occupancy measurement across wide bands with significant noise level changes.

## References

- Bello, N. and Ogbeide, K. O. (2021). Design of a real-time swept spectrum analyser with USRP B210. *Nigerian Journal of Enviromental Science and Technology*, 5(2), pp. 329-339.
- Ephraim, Y. and Malah, D. (1985). Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE transactions on acoustics, speech, and signal processing*, 33, pp. 443–445.
- Malah, D., Cox, R. V. and Accardi, A. J. (1999). *Tracking speech-presence uncertainty to improve speech enhancement in non-stationary noise environments*. s.l., s.n., pp. 789–792.
- Martin, R. (2001). Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Transactions on speech and audio processing*, 9, pp. 504–512.

- Martin, R. (2006). Bias compensation methods for minimum statistics noise power spectral density estimation. *Signal Processing*, 86, pp. 1215–1229.
- Mateo, C. and Talavera, J. A. (2018). Short-time Fourier transform with the window size fixed in the frequency domain. *Digital Signal Processing*, 77, pp. 13–21.
- Mauler, D. and Martin, R. (2006). *Noise power spectral density estimation on highly correlated data*. s.l., s.n.
- McCree, A. et al., (1996). A 2.4 kbit/s MELP coder candidate for the new US Federal Standard. s.l., s.n., pp. 200–203.
- Qu, S.-M., Shi, Q.-L. and Liu, X. (2014). *Study on Calculation Method of Spectrum Occupancy Threshold Based on Cognitive Radio Access*. s.l., s.n., pp. 65–71.
- Seeber, B. (2014). GNU Radio Tutorials. *Ettus Research*.
- Series, S. M. (2017). Spectrum occupancy measurements and evaluation. *International Telecommunication Union*.
- Turner, C. S. (2012). Johnson-nyquist noise. url: <http://www.claysturner.com/dsp/Johnson-NyquistNoise>.

---

**Cite this article as:**

Bello N. and Ogbeide K. O., 2022. Spectral Noise Estimation: A Python 3 Implementation of the Minimum Statistics Estimation. *Nigerian Journal of Environmental Sciences and Technology*, 6(1), pp. 1-12. <https://doi.org/10.36263/nijest.2022.01.0300>